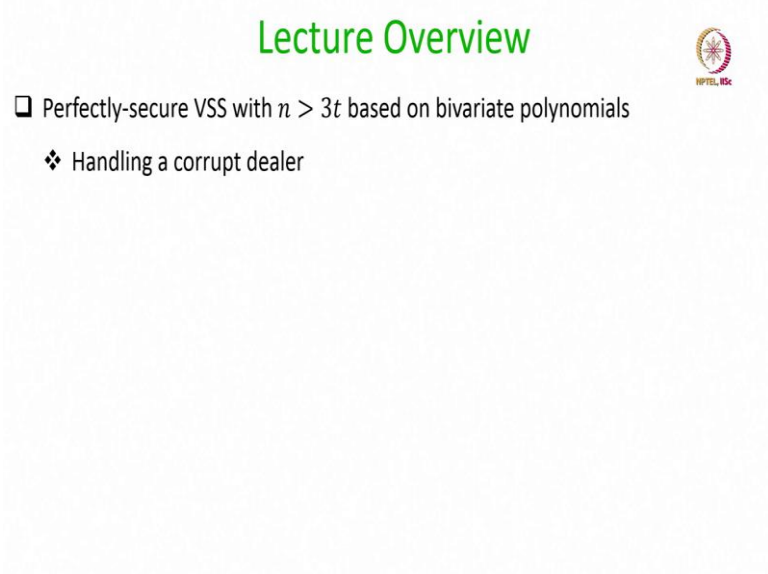


Secure Computation: Part II
Prof. Ashish Choudhury
Department of Computer Science and Engineering
Indian Institute of Science, Bengaluru

Lecture - 39
Perfectly-Secure VSS with n Greater than $3t$ Part II

Hello everyone, welcome to this lecture.

(Refer Slide Time: 00:25)



The slide is titled "Lecture Overview" in green text. It contains two bullet points: a square icon followed by "Perfectly-secure VSS with $n > 3t$ based on bivariate polynomials" and a diamond icon followed by "Handling a corrupt dealer". In the top right corner, there is a circular logo with a star and the text "NPTEL IISc" below it.

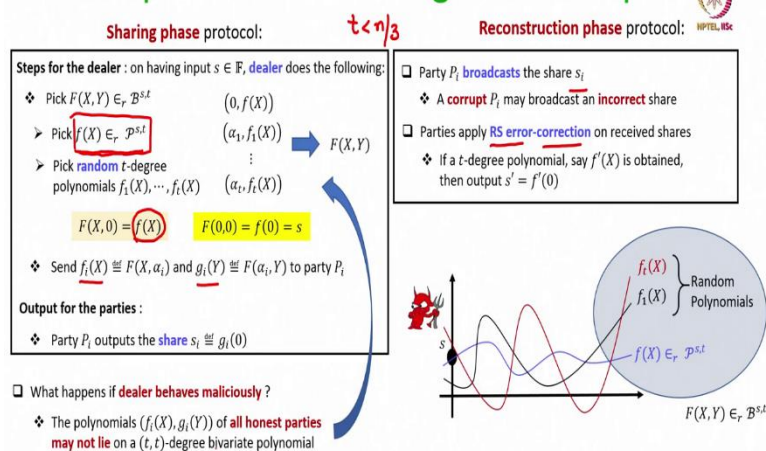
Lecture Overview

- Perfectly-secure VSS with $n > 3t$ based on bivariate polynomials
- ❖ Handling a corrupt dealer

So, the plan for this lecture is as follows. In this lecture, we will see how we can modify the perfectly secure VSS scheme that we had discussed in the last lecture based on bivariate polynomials and make it handle even a potential corrupt dealer.

(Refer Slide Time: 00:42)

The Simpler VSS Scheme Against Corrupt Dealer



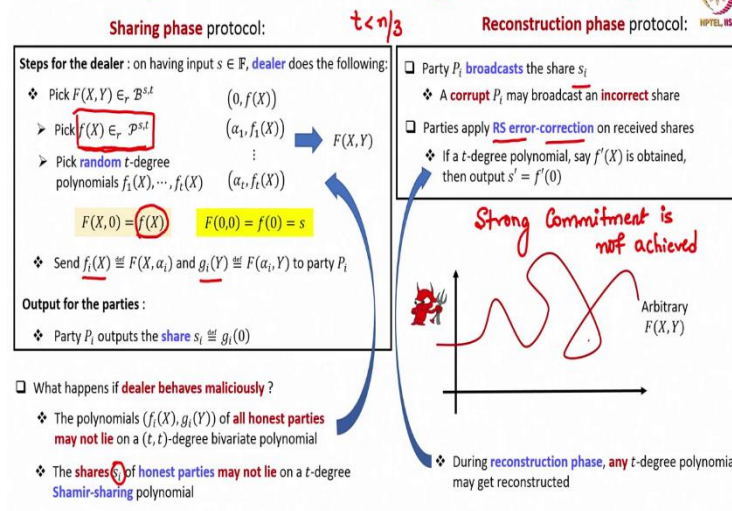
So, just to recap, this was the simpler verifiable secret sharing scheme, where we assumed for simplicity that the dealer is honest and the sharing phase protocol is the following. Dealer first picks Shamir sharing polynomial of degree t whose constant term will be the secret. And it then embeds this Shamir sharing polynomial in a random bivariate t degree polynomial. And the i th party is provided the i th row polynomial and the i th column polynomial on that bivariate polynomial.

Each party outputs the constant term of the received column polynomial as its share. And, in the reconstruction protocol, every party makes public its share, the corrupt parties may provide incorrect shares. So, we apply the Reed-Solomon error correction. And, since we are working in the setting $t < \frac{n}{3}$ and the shares s_i lie on a t degree polynomial $f(X)$. The parties can apply the Reed-Solomon error correction and reconstruct back the Shamir sharing polynomial $f(X)$.

They can then output the constant term of that polynomial as the secret. We have also discussed in the last lecture what goes wrong in this protocol if the dealer behaves maliciously. So, remember that up to t parties are allowed to be corrupt, and this possibly includes the dealer as well. So, if the dealer behaves maliciously, then what it can do is that it may distribute inconsistent polynomials to the honest parties.

(Refer Slide Time: 02:44)

The Simpler VSS Scheme Against Corrupt Dealer

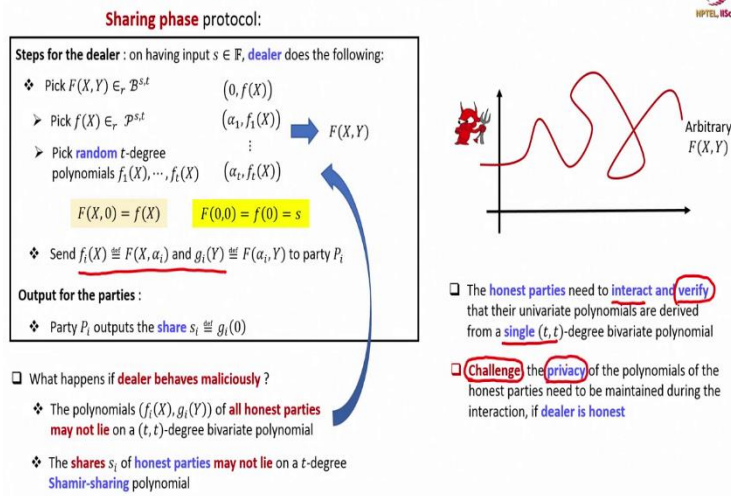


Namely, the f and the g polynomials given to the honest parties may be now arbitrary polynomials and may not lie on a t degree bivariate polynomial. As a result of this what can go happen is that the shares of the honest parties may not lie on a t degree polynomial. This will have further consequences in the reconstruction phase protocol because Reed-Solomon error correction may not work properly. And hence, any t degree polynomial may get reconstructed and hence, any secret may get reconstructed.

So, if the dealer behaves maliciously in this protocol, then the strong commitment property is violated. So, the strong commitment is not achieved. And now our goal will be to take this sharing phase protocol and incorporate additional steps here, additional mechanisms here to ensure that even a potentially corrupted dealer is following the protocol instructions properly and distributing consistent polynomials to the honest parties.

(Refer Slide Time: 03:53)

The Simpler VSS Scheme Against Corrupt Dealer



So, now this will require interaction among the parties to verify whether their f and g polynomials lie on a single bivariate polynomial. And that is why this scheme, or this protocol, is called as a verifiable secret sharing, because the parties can verify. One simple way to verify whether the polynomials of all the honest parties are derived from a single bivariate polynomial will be to ask each party to make public the received f and g polynomials.

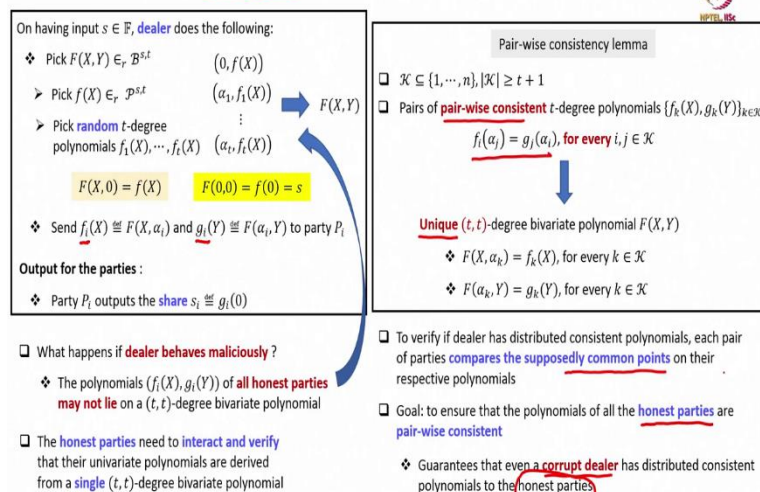
And, then we can do some consistency checks and find out whether the polynomials are derived from a single bivariate polynomial or not, but that will simply breach the privacy. So, we require here a mechanism which allows the parties to verify whether their f and g polynomials are derived from a single bivariate polynomial. And that too ensuring the privacy of their polynomials if the dealer would have been honest right. So, that is the challenging part here.

So, it is like two conflicting goals need to be achieved. We want the verification mechanism during the sharing phase protocol to verify whether the f and g polynomials of all the honest parties are derived from a single t degree bivariate polynomial. And, at the same time during this verification mechanism, we would like that if the dealer is honest, then nothing about the row and column polynomials of the honest parties should be revealed during the verification mechanism.

Because we would like to maintain the privacy property for the verifiable secret sharing if the dealer would have been honest.

(Refer Slide Time: 05:41)

VSS Scheme: Checking the Consistency of the Polynomials



So, how do we check whether a potentially corrupt dealer has distributed consistent row and column polynomials to the honest parties? And, to do that what we are going to do is we are going to invoke the pair wise consistency lemma. So, we will have some steps in the protocol which will depend on or which will utilize the pair wise consistency lemma. So, let us quickly go through the pair wise consistency lemma statement.

So, the lemma states that if you have $t + 1$ or more number of row and column polynomials which are pairwise consistent, then all those row and column polynomials lie on a single t degree bivariate polynomial. And it is precisely this property which the parties are going to utilize while doing the verification. So, what the parties are going to now do is the following.

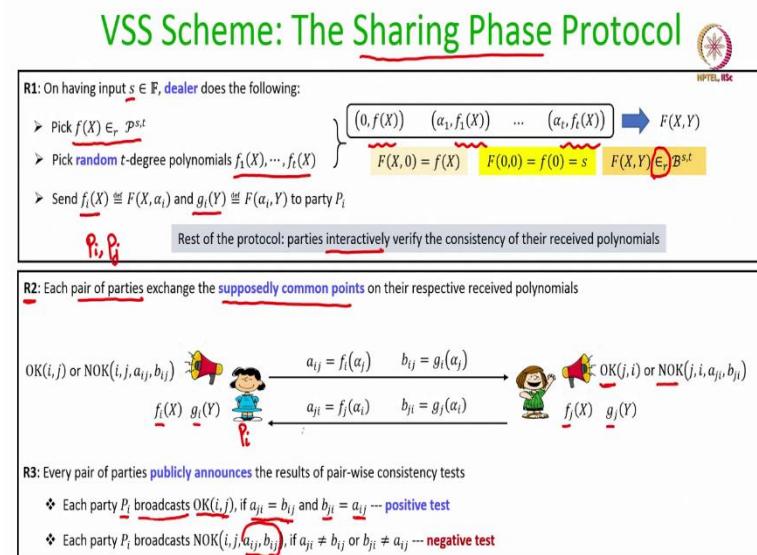
Once dealer distributes the row and column polynomials to their respective parties, each pair of parties will compare the supposedly common points on their respective polynomials and publicly announce the result. And, then the goal will be to ensure that all the honest parties, their polynomials are pairwise consistent. If they are not pairwise consistent and there will be publicly raised disputes and then dealer has to come into picture and then dealer has to resolve those disputes.

And, after end of all those things, it will be ensured that the polynomials of all honest parties their row and column polynomials are pairwise consistent. And that will automatically ensure that even if the dealer is corrupt, it has distributed row and column polynomials derived from a single bivariate polynomial to all the honest parties.

I would like to stress here that here we would like to check the pair wise consistency between the row and column polynomials of only the honest parties. We do not care what exactly the corrupt parties get because they can output anything. So, the strong commitment property is only with respect to the shares of the honest parties. We do not care what shares the corrupt parties output because, they may not follow the protocol instructions.

In fact, they can output any share whatever they want at the end of the sharing phase protocol. So, that is the main idea here. But now, the obvious challenges are that how do we ensure that the parties verify the pair wise consistency of their respective polynomials and that too guaranteeing the privacy of the polynomials if the dealer is honest.

(Refer Slide Time: 08:37)



So, here is how the protocol proceeds. The round 1 of the protocol, the sharing phase protocol to be more specific will be the following. On having the secret s dealer will pick a random t degree bivariate polynomial whose constant term is the secret s . For doing that, it will first pick a random Shamir sharing polynomial whose constant term is the secret s and then it will pick t additional t degree univariate polynomials in X .

And, then it will try to interpolate this $t + 1$ number of X univariate polynomials which will guarantee that the constant term of the bivariate polynomial is the secret of the dealer. And, if the dealer is honest and if he picks the bivariate polynomial like this, that also ensures that the resultant bivariate polynomial is a random bivariate polynomial whose constant term is the dealer's secret.

Then, as done in the previous case where we assume the dealer is honest, dealer will give the i th row polynomial and the i th column polynomial on this bivariate polynomial to the i th party. And now the rest of the protocol steps will require interaction among the parties to verify whether the dealer has distributed consistent row and column polynomials to the honest parties.

Because, if the dealer is not honest and it may not follow the protocol instructions, it may distribute arbitrary polynomials to the honest parties. So, once the dealer distributes the row and column polynomials to the respective parties, the round 2 starts. Just to recall, we assume here that the protocol executes as a sequence of communication rounds wherein each round parties perform some computation, decides what message to send to its neighbor, what message to broadcast.

It broadcast those messages and it receives the messages which have been sent by its neighbor and then it goes to the next round. Also for simplicity we will assume that if an expected message is not coming from a designated sender party, then the receiver party in that round substitutes that expected message by some default message and proceed to the next round or next steps.

We will not write separate steps that ok, if this message which is expected to arrive and it does not arrive what should we do and so on. So, in round 2 each pair of parties will now perform the pair wise consistency of their received polynomials by exchanging the supposedly common points on their received polynomials.

So, every pair of parties P_i and P_j , where P_i and P_j may be the same parties or they may be distinct parties do the following. So, the i th party has its row and column polynomial and the j th party has its row and column polynomial. The i th party P_i will send the j th point on its row and column polynomial to the party P_j .

And, in the same round in parallel, the j th party will be sending the i th point on its row and column polynomial to party P_i . Now, ideally if dealer is honest then we expect that the j th point on i th row polynomial should be same as i th point on the j th column polynomial and vice versa.

And, that is what each pair of parties P_i and P_j checks during round 3. So, once these supposedly common points are exchanged, every pair of parties during the round 3 publicly announces the result of the pair wise consistency test. So, let us see what party P_i does. So, it checks whether the i th point on the j th column polynomial which it has received from the party P_j matches the j th point on its down column polynomial.

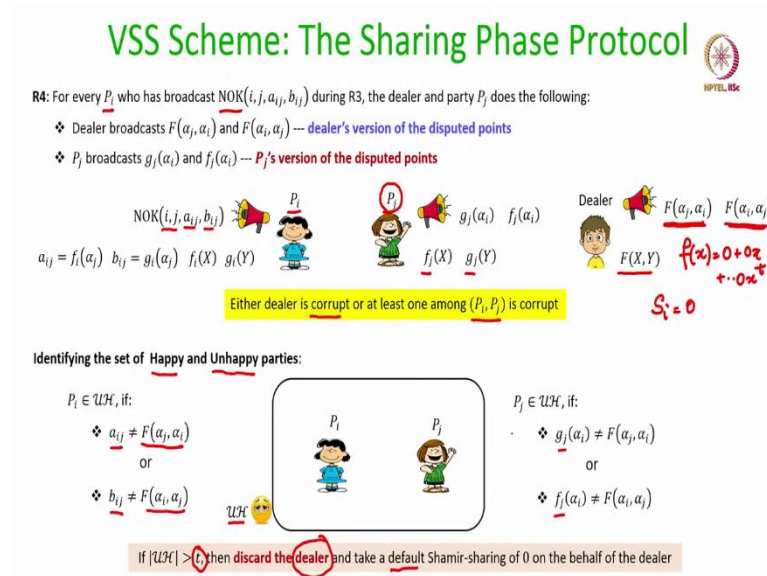
And, it also checks whether the i th point on the j th parties column polynomial which it has received is same as the j th point on its own row polynomial. If both these tests pass, then it says it makes public an OK message; saying that “I am fine with j ”; that means, “My polynomials are pair wise consistent with j th party’s polynomial” and it broadcasts an OK message. So, recall that we assume that we have a system wide broadcast channel available.

So, if party P_i is broadcasting an OK message, it will be delivered identically to all the honest parties. How do we realize such a system wide broadcast channel? Well, we can emulate the effect of broadcast by running any reliable broadcast protocol. Whereas, if any of this pair wise consistency check fails for P_i , then it makes public and NOK message, indicating that it is not ok with P_j or; that means, it is in conflict with P_j .

And, it will make public the disputed points. Namely, its own version of the disputed points which indicates a negative test ok, that is a round 3. P_j will be performing a similar check and accordingly, in the same round 3, P_j might be either broadcasting an OK message for P_i or it might be broadcasting an NOK message for P_i .

It might be possible that even though the dealer is honest, one of these parties P_i and P_j is corrupt and they unnecessarily broadcast an NOK message. But what we can conclude is that if the dealer P_i and P_j are all honest, then P_i will be broadcasting an OK message for P_j and P_j will be broadcasting NOK message for P_i .

(Refer Slide Time: 15:01)



Now, let us see what happens during the round 4 of the protocol. So, in the round 4 of the protocol, for every party P_i who has broadcasted an NOK message for P_j during round 3; the dealer and the party P_j makes public their version of the disputed points. So, imagine that during round 3, there is some party P_i who has broadcasted an NOK message against P_j , indicating that P_i has performed a pair wise consistency test.

And the pair wise consistency test is negative from the viewpoint of P_i , that is why it is saying that I am not fine with P_j 's polynomial. My polynomials are mismatching with P_j 's polynomial. So, for every such complaint raised against the dealer or against the party P_j to be more specific by P_i , dealer makes public its own version of the disputed points. So, what are the disputed points here? The j th point on the i th row polynomial and the j th point on the i th column polynomial.

So, dealer has those points because the dealer only has distributed those points to party P_i because, it has the full bivariate polynomial. So, dealer now makes public its version of the disputed points. And, in the same round since the dispute is against P_j , P_j makes public its version of the disputed points which P_j would have received from the dealer during round 1. So, P_j has its own row and column polynomial. A dispute has been raised by P_i during round 3.

So, P_j now goes and makes it stand public during the round 4 by making public its own version of the corresponding disputed point. Now, what can we conclude here? If at all there is a dispute between P_i and P_j , then either the dealer is corrupt and it has distributed inconsistent polynomials to P_i , P_j . Or if the dealer is honest, but still if there is a dispute between and between a P_i and P_j then at least one of them is corrupt.

Because, if the dealer is honest, if P_i is honest and if P_j is also honest all these three parties are honest, then there then the polynomial, then the row and column polynomials of P_i and P_j will be pairwise consistent. And there will be no dispute in the first place. Namely, P_i will say OK for P_j and P_j will also say OK for P_i . So, if at all there is a dispute between P_i and P_j , then that implies that either the dealer is corrupt or one of the two parties P_i or P_j is corrupt.

So, for every such dispute, dealer, and the corresponding party against whom the complaint has been raised, have to come into picture during round 4. And, they have to make public their version of the corresponding disputed points. Now, based on what the party P_j and the dealer makes public during round 4, we identify the set of happy and unhappy parties. Now, who will be the unhappy parties?

We will say that party P_i is unhappy if it has raised a complaint against P_j during round 3. And, corresponding to that complain during round 4, dealer has made public some points which does not match P_i 's version of the disputed points right. So, recall that P_i has made public its version of the disputed points during round 3 and P and dealer has made now public, dealer's version of the disputed points during round 4.

So, if dealer is not taking P_i 's side, what does that mean? If the dealer's version of the points, disputed points does not match any of the P_i 's version of the disputed points, then that will be taken as if dealer is not taking P_i 's side. And that implies that P_i will be considered as an unhappy party. And, in the same way P_j will be considered an unhappy party, if the dealer's version of the disputed points does not match P_j 's version of the disputed points which P_j has made public during the round 4.

It might be possible that dealer's version of the disputed points matches P_i 's version and P_j 's version in which case none of them are unhappy, that could be possible. But, if at all

it so, happens that the dealer's version of the disputed points does not match either the P_i 's versions or the P_j 's version, then the corresponding party is considered as an unhappy party.

And notice that this set of happy and unhappy parties will be publicly known, because the set of happy and unhappy parties are decided based on the NOK messages. And, what the dealer and the corresponding complainant makes public during round 4 and everything, all the disputed points are made public through broadcast channel.

So, everyone will know which party has made which points has the disputed points, what is their respective version, what is the dealer's version, whether they are matching the complainer's version or the complainant's version. And depending upon that the set of unhappy parties will be publicly decided.

Now, before we proceed further as the sanity check at the end of round 4, once the set of unhappy parties is decided, parties check whether there are more than t unhappy parties. So, this \mathcal{UH} denotes unhappy parties. And the parties who are not in this set \mathcal{UH} , they will be considered as happy parties.

So, if there are more than t unhappy parties then that automatically implies that the dealer is corrupt. And that is why it is safe to terminate and halt the sharing phase protocol at this point itself and discard the dealer. Because, if the dealer is honest then no honest party will be present in the unhappy set. It will be only the corrupt parties who will be present in the unhappy set.

Because the polynomials of all honest parties will be pairwise consistent and will lie on a single bivariate polynomial. So, if at all the number of unhappy parties is more than t , namely the number of maximum corruptions in the system it automatically implies that the dealer is corrupt. And, hence, they can discard the dealer and they can take a default Shamir sharing of 0, as the sharing on the behalf of the dealer.

So, what does this default Shamir sharing of 0 mean? Every party sets its share as 0, if the sharing polynomial $f(X)$ is a t degree polynomial, where the constant term is 0. And all other coefficients are also 0, that is a default Shamir sharing of 0.

(Refer Slide Time: 22:39)

VSS Scheme: The Sharing Phase Protocol

RS: For every unhappy party $P_i \in \mathcal{UH}$, the dealer and every happy party $P_j \in \mathcal{UH}$ does the following :

- ❖ Dealer broadcasts the polynomials $F(X, \alpha_i)$ and $F(\alpha_i, Y)$
- ❖ P_j broadcasts $g_j(\alpha_i)$ and $f_j(\alpha_i)$

Handwritten annotations in red:

- $F(x, \alpha_i) \equiv$
- $F(\alpha_i, y)$
- $P_i \in \mathcal{UH}$
- P_i

Diagram showing a box labeled P_i with a sad face icon, and a person icon labeled Dealer with $F(X, Y)$ next to it. A yellow smiley face icon is labeled \mathcal{UH} .

Now, if the dealer is not discarded during round 4, what does that mean? It means that there are at most t unhappy parties. So, the protocol is not yet over because it could be possible that dealer is corrupt and there are some honest parties who are still present in the unhappy set. We must ensure that they also get their respective row and column polynomials.

So, for that we have this 5th round in the protocol, round 5 of the protocol where for every unhappy party P_i dealer makes public the supposedly row and common column polynomials of that unhappy party. Well, if the dealer is honest then indeed its row and column polynomials which dealer is making public for those unhappy parties P_i will be the correct polynomials, but it is not necessary that dealer is honest.

What dealer may simply do is for every unhappy party, it may simply broadcast an arbitrary row and column polynomial which is now not consistent with the row and column polynomials of the happy parties right that could be possible. So, to ensure, to enforce that even a potentially corrupt dealer broadcasts the correct row and column polynomials for every unhappy party, the happy parties have to do something.

So, the happy parties namely the parties who are not in the unhappy set, they do the following in parallel during round 5. They make public the supposedly common points on their row and column polynomials which should lie on the row and column polynomials of every unhappy party right. So, what is happening here is that suppose dealer is there

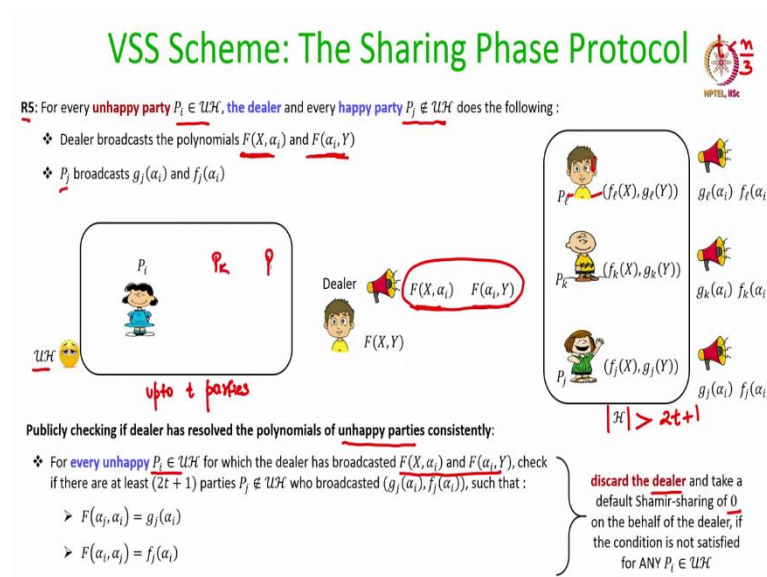
and thus this party P_i is an unhappy party. So, this unhappy party P_i is supposed to now receive its row and column polynomial.

Because whatever row and column polynomial it would have received from the dealer during round 1 is not consistent with the parties with the remaining parties. So, now what the dealer is may doing in the round 5 is we are asking dealer fine dealer, you make now public the row and column polynomials of this party P_i ; he is still unhappy. So, dealer is saying ok its row and its row polynomial should be $F(X, \alpha_i)$ and its column polynomial should be $F(\alpha_i, Y)$.

But what is the guarantee that these are indeed the correct row and column polynomials? So, what we are asking now here is that every happy party P_j for whom it has been ensured that they are pair wise, their polynomials and pair wise consistent, lie on a single bivariate polynomial. They make public, they are supposedly common points on this row and column polynomials.

Because, they also have two points which lie which should supposedly lie on the row and column polynomials corresponding to this unhappy party P_i . So, P_j makes public those points. Now, it is not necessary that every P_j who is a happy party, who belongs to the happy set, broadcasts the correct points on P_i 's row and column polynomials. They may be corrupt as well. So, what now we are doing is we are going to do the following.

(Refer Slide Time: 26:21)



So, say we have an unhappy party P_i , for that dealer is now making public the i th row and column polynomial public and we have a bunch of happy parties right. So, remember we are working with the setting $t < \frac{n}{3}$; that means, the number of happy parties if the dealer is not discarded is at least $2t + 1$. Now, what each of these happy parties is doing is that they are making public the supposedly common points on the i th party's row and column polynomials which they have.

And now we have to check the following. We have to publicly check whether the dealer has resolved the polynomials of unhappy parties consistently; that means, whether the dealer has made public the right polynomials on the behalf of every unhappy party. How do we check that? For that, we check that for every unhappy party P_i for whom dealer has made public the row and column polynomial, corresponding to that whether there exist at least $2t + 1$ happy parties who have broadcasted the corresponding points which lies on the polynomials broadcasted by the dealer.

If this condition is not satisfied, then again, we can show that it implies that the dealer is corrupt. And hence it is safe to discard the dealer and terminate the sharing phase protocol here itself, by asking every party to output the share 0 on the behalf of the dealer. However, if this condition is satisfied for every unhappy party P_i right. So, remember there could be up to t parties in this set, in this set of unhappy parties. There could be a party P_k as well.

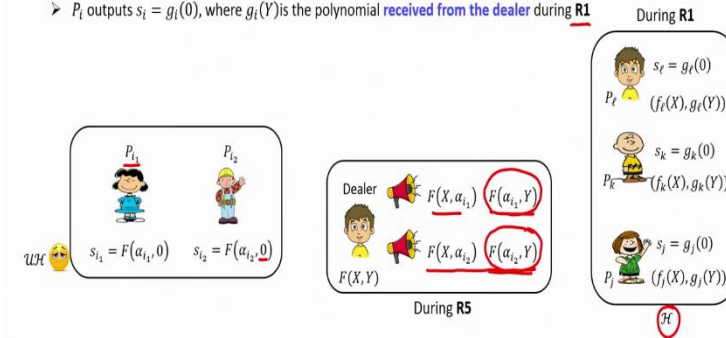
There could be parties $P_{i_1}, P_{i_2}, P_{i_3}$ and so on. There could be up to t such parties. So, the condition that I have mentioned here, it should hold for every unhappy party P_i . Namely, for every such unhappy party P_i whatever polynomial dealer is making public, it should be consistent with at least $2t + 1$ points which are made public by the happy parties. If the condition is not satisfied, safely discard the dealer.

(Refer Slide Time: 29:16)

VSS Scheme: The Sharing Phase Output Decision

Output decision: If the dealer is **not discarded** throughout the protocol, then every P_i outputs its **shares** as follows

- ❖ If $P_i \in \mathcal{UH}$
 - P_i outputs $s_i = F(a_i, 0)$, where $F(a_i, Y)$ is the polynomial **broadcasted by the dealer during R5**
- ❖ If $P_i \notin \mathcal{UH}$
 - P_i outputs $s_i = g_i(0)$, where $g_i(Y)$ is the polynomial **received from the dealer during R1**



Now, let us see the output of the sharing phase protocol. So, if the dealer is not discarded then every party P_i sets its share as follows. So, there could be two possibilities depending upon whether the party P_i is finally ending up being a part of the happy set or unhappy set. If it ends up in the unhappy set, then on the behalf of such P_i , dealer would have made public the row and column polynomials during the round 5 of the protocol.

So, what P_i does is it takes the column polynomial which dealer has now made public during round 5. It forgets whatever polynomial such P_i has received from the dealer during round 1. And, now the constant term of this new column polynomial which has been made public by the dealer during round 5 is taken as the share by party P_i . So, if there is this party P_{i_1} corresponding to that, dealer would have made public these two polynomials.


So, what party P_{i_1} does is it takes this column polynomial and output the constant term. Similarly, if P_{i_2} is another unhappy party and for that dealer would have made public these two polynomials. P_{i_2} would take will now take this column polynomial and output the constant term as its share.

Whereas all the happy parties, they output the constant term of the column polynomials which they have received from the dealer during round 1; because those polynomials have been found to be pairwise consistent by the happy parties. So, if P_ℓ is a part of the happy

set, it outputs the constant term of the ℓ th column polynomial which it has received from the dealer during round 1. And similarly other parties in the happy set obtain their shares.

(Refer Slide Time: 31:20)

References



- ❑ Rosario Gennaro, Yuval Ishai, Eyal Kushilevitz, Tal Rabin: The round complexity of verifiable secret sharing and secure multicast. STOC 2001: 580-589
- ❑ Anirudh Chandramouli, Ashish Choudhury, Arpita Patra: A Survey on Perfectly-Secure Verifiable Secret-Sharing. ACM Computing Surveys, 2022

So, that ends the description of the sharing phase protocol. In the next lecture, we will do a rigorous analysis of this sharing phase protocol and see whether it satisfies the strong commitment property against the potentially corrupted dealer and whether it maintains the correctness and privacy property, that we have achieved for the VSS, simpler VSS scheme assuming an honest dealer. So, these are the references used for today's lecture.